Journal of Computational and Applied Mathematics 233 (2010) 2867–2875

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Exponentials of skew-symmetric matrices and logarithms of orthogonal matrices

João R. Cardoso [a,b,*], F. Silva Leite [c,b]

[a] Institute of Engineering of Coimbra, Rua Pedro Nunes, 3030-199 Coimbra, Portugal
[b] Institute of Systems and Robotics, University of Coimbra, Pólo II, 3030-290 Coimbra, Portugal
[c] Department of Mathematics, University of Coimbra, 3001-454 Coimbra, Portugal

ARTICLE INFO

Article history:
Received 11 July 2008

MSC:
65Fxx

Keywords:
Skew-symmetric matrix
Orthogonal matrix
Matrix exponential
Matrix logarithm
Scaling and squaring
Inverse scaling and squaring
Padé approximants

ABSTRACT

Two widely used methods for computing matrix exponentials and matrix logarithms are, respectively, the scaling and squaring and the inverse scaling and squaring. Both methods become effective when combined with Padé approximation. This paper deals with the computation of exponentials of skew-symmetric matrices and logarithms of orthogonal matrices. Our main goal is to improve these two methods by exploiting the special structure of skew-symmetric and orthogonal matrices. Geometric features of the matrix exponential and logarithm and extensions to the special Euclidean group of rigid motions are also addressed.

1. Introduction

Given a square matrix $X \in \mathbb{R}^{n\times n}$, the exponential of X is given by the absolute convergent power series

$$e^X = \sum_{k=0}^{\infty} \frac{X^k}{k!}.$$

Reciprocally, given a nonsingular matrix $A \in \mathbb{R}^{n\times n}$, any solution of the matrix equation $e^X = A$ is called a *logarithm* of A. In general, a nonsingular real matrix may have an infinite number of real and complex logarithms. However, if A has no eigenvalues on $\mathbb{R}_0^-$ then there exists a unique real logarithm of A whose eigenvalues lie on the open strip $\{z \in \mathbb{C} : -\pi < \operatorname{Im} z < \pi\}$ of the complex plane [1, ch. 1]. This unique logarithm is the one we are interested in this work. Hereafter, it will be referred as the principal logarithm (or just as the logarithm) of A and denoted by $\log A$.

Exponentials of skew-symmetric matrices and logarithms of orthogonal matrices arise in many areas of engineering and control theory, namely in problems related with dynamical systems and rigid body dynamics (see, for instance, [2–5]).

The algebra of skew-symmetric matrices is denoted by

$$so(n) = \{S \in \mathbb{R}^{n\times n} : S^\top = -S\}$$

and the group of orthogonal matrices by

$$O(n) = \{Q \in \mathbb{R}^{n\times n} : Q^\top = Q^{-1}\}.$$

* Corresponding author at: Institute of Engineering of Coimbra, Rua Pedro Nunes, 3030-199 Coimbra, Portugal.
E-mail addresses: jocar@isec.pt (J.R. Cardoso), fleite@mat.uc.pt (F.S. Leite).

0377-0427/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2009.11.032

The eigenvalues of skew-symmetric and orthogonal matrices allow nice representations. If λ is an eigenvalue of $S \in so(n)$ then $\lambda = \alpha\,\mathrm{i}$, for some $\alpha \in \mathbb{R}$; if λ is an eigenvalue of $Q \in O(n)$, then $\lambda = \mathrm{e}^{\theta\,\mathrm{i}}$, for some $\theta \in \mathbb{R}$. The orthogonal matrices with determinant one also form a group, called the special orthogonal group or group of rotations:

$$SO(n) = \{Q \in \mathbb{R}^{n\times n} : \ Q^\top = Q^{-1}, \ \det(Q) = 1\}.$$

There is an important connection between $so(n)$ and $SO(n)$ by means of the exponential and logarithm functions: the exponential of a matrix in $so(n)$ belongs to $SO(n)$ and every special orthogonal matrix has a skew-symmetric logarithm. The latter property reflects the surjectivity of the exponential mapping and plays an important role in the description of rotational motions. In Section 2 we point out some geometric issues concerning to exponentials and logarithms of rotation matrices.

The major part of this paper is devoted to the computation of the exponential of a skew-symmetric matrix and the principal logarithm of an orthogonal matrix with no negative eigenvalues. We provide effective methods for performing these computations. In Section 3 we analyze the behavior of the classical scaling and squaring method with diagonal Padé approximants (see [6,7] and the references therein), when applied to the particular case of skew-symmetric matrices. Our aim is to show that this classical method provides an effective way to compute the exponential of these matrices. By exploiting the special properties of the matrices in $so(n)$, we are able to obtain the exact Padé error and minimize the computational cost of the method. Similarly, we proceed in Section 4 with the problem of computing the principal logarithm of an orthogonal matrix. We base our analysis on previous work developed in [8] and improve the well-known inverse scaling and squaring method (combined with Padé approximation) when restricted to orthogonal matrices. In particular, we show that two square roots suffice to guarantee a Padé error close to the unit roundoff in IEEE double precision arithmetic ($u \approx 1.1 \times 10^{-16}$).

One important feature of the methods based on diagonal Padé approximants is that they are structure preserving (in exact arithmetic), that is, they respect the properties of the exponential and logarithm functions: diagonal Padé approximants of the exponential of a skew-symmetric matrix produce special orthogonal matrices and diagonal Padé approximants to the logarithm of a matrix in $SO(n)$ belong to $so(n)$.

The configuration space for rigid body motions in $\mathbb{R}^n$ is the special Euclidean group, defined by

$$SE(n) = \left\{ \begin{bmatrix} Q & \mathbf{u} \\ \mathbf{0} & 1 \end{bmatrix} : Q \in SO(n), \mathbf{u} \in \mathbb{R}^{n\times 1} \right\}.$$

While $SE(n)$ describes configurations, its Lie algebra $se(n)$, defined by

$$se(n) = \left\{ \begin{bmatrix} S & \mathbf{v} \\ \mathbf{0} & 0 \end{bmatrix} : S \in so(n), \mathbf{v} \in \mathbb{R}^{n\times 1} \right\},$$

measures velocities. The case $n = 3$ is particularly important in Robotics, since the configuration of a robot can be described by rotations and translations in $\mathbb{R}^3$ (see, for instance, [9,2]). Elements in $se(3)$ are called "twists" in the Robotics literature. In Section 5 we make some remarks on the computation of exponentials and logarithms of matrices belonging to $se(n)$ and $SE(n)$, respectively.

Throughout the paper we use the 2-norm $\|.\|$ of a matrix, which is given by its largest singular value.

2. Geometric aspects of the exponential and logarithm

To any vector $\mathbf{u} = (u_1, \ u_2, \ u_3) \in \mathbb{R}^3$ one can associate the 3×3 skew-symmetric matrix

$$S_{\mathbf{u}} = \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix}.$$

It is easy to see that, for every $\mathbf{v} \in \mathbb{R}^3$,

$$\mathbf{u} \times \mathbf{v} = S_{\mathbf{u}}\,\mathbf{v},$$

where $\times$ stands for the cross product.

Now consider the following problem: given a unit vector $\mathbf{u} \in \mathbb{R}^3$ and an angle $\theta \in \mathbb{R}$, find the rotation matrix R that rotates any vector through the angle θ about an axis with direction $\mathbf{u}$. The matrix exponential gives the elegant solution:

$$R = \mathrm{e}^{S_{\mathbf{u}}\theta}.$$

The exponential of a skew-symmetric 3×3 matrix may be computed by means of the well-known Rodrigues formula

$$\mathrm{e}^{S_{\mathbf{u}}\theta} = I + \sin\theta\, S_{\mathbf{u}} + (1 - \cos\theta)\, S_{\mathbf{u}}^2.$$

Conversely, given $R \in SO(3)$ (with no negative eigenvalues) consider the problem of finding the axis direction $\mathbf{u}$ and the angle θ of rotation. Using the matrix exponential, we can formulate this problem as follows: determine a unitary vector $\mathbf{u}$ and $\theta \in]-\pi, \pi[$ such that

$$R = \mathrm{e}^{S_{\mathbf{u}}\theta}.$$

The matrix logarithm provides the simple answer

$$S_{\mathbf{u}}\,\theta = \log(R), \tag{1}$$

or equivalently,

$$S_{\mathbf{u}} = \frac{1}{\theta}\,\log(R),$$

whenever $\theta \neq 0$. This means that the direction of the rotation axis is given by the logarithm of the matrix associated with $\mathbf{u}$. Moreover, taking norms in (1) yields

$$\| S_{\mathbf{u}} \|\,|\theta| = \| \log(R)\|,$$

and, since $\| S_{\mathbf{u}} \| = 1$, the angle of rotation is related with the norm of $\log(R)$ by

$$|\theta| = \| \log(R)\|.$$

This relationship between skew-symmetric and rotation matrices by means of exponentials and logarithms are the key to explain the importance of these matrix functions in robotics [9,2]. There are many other geometric problems involving exponentials and logarithms of matrices; the references [10,3–5] include some applications to the design of trajectories for rigid bodies in the 3D space.

A widely used formula for computing the logarithm of a 3×3 rotation matrix is

$$\log R = \frac{\theta}{2\sin\theta}(R^\top - R),$$

where θ satisfies $1 + 2\cos\theta = \mathrm{trace}(R)$, $\theta \neq 0$, $-\pi < \theta < \pi$. When $\theta = 0$ one has the trivial case $R = I$ and $\log R = 0$.

3. Computing the exponential in $so(n)$

In this section we consider the problem of computing the exponential of a general $n \times n$ skew-symmetric matrix. It is worth emphasizing that this topic has already been addressed in [11], where the authors proposed a generalization of the Rodrigues formula for a general n, yet they have not provided implementing issues. Although the proposed method is interesting from a theoretical point of view, its computational cost seems to be prohibitive, unless n is small. It involves at least the solution of a $pn^2 \times pn^2$ linear system, with p being the number of distinct pairs of complex eigenvalues.

The diagonal Padé approximant of order m of e^x, $x \in \mathbb{C}$, is the rational function

$$r_m(x) = \frac{p_m(x)}{p_m(-x)},$$

with

$$p_m(x) = \sum_{k=0}^{m} \frac{(2m-k)!m!}{(2m)!k!(m-k)!}x^k.$$

It satisfies the relationship

$$e^x - r_m(x) = \mathcal{O}(x^{m+1}),$$

which means that r_m gives accurate results only when x lies in a small neighborhood of the origin. Padé approximants inherit many properties of the exponential function. For instance, $[r_m(x)]^{-1} = r_m(-x)$.

In the matrix case, we can write

$$e^A \approx r_m(A) = p_m(A)\,[p_m(-A)]^{-1},$$

for any $A \in \mathbb{R}^{n\times n}$ with norm sufficiently close to zero.

The standard scaling and squaring technique, combined with Padé approximants, may be described by the following three steps (see [12,7]):

1. Find a positive integer k such that $\frac{\|A\|}{2^k} \leq \delta < 1$ (δ is a tolerance).
2. Evaluate $r_m\left(\frac{A}{2^k}\right)$.
3. $e^A \approx \left[r_m\left(\frac{A}{2^k}\right)\right]^{2^k}$.

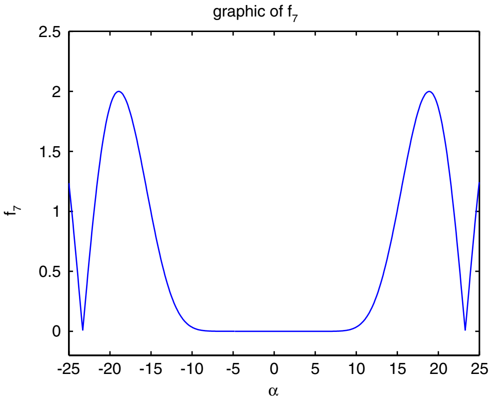

Fig. 1. Graphic of the function $f_7(\alpha)$.

The numbers k in step 1 and m in step 2 must guarantee the accuracy in step 3. There are several upper bounds for the Padé error proposed in the literature (see, for instance, [13,14]). Unfortunately, when these bounds give weak estimates we may have an overscaling that leads to the computation of unnecessary squarings.

Assuming that S is a skew-symmetric matrix, $r_m(S)$ is a matrix function because the polynomial p_m does not have zeros at the eigenvalues of S [15]. Moreover the eigenvalues of $r_m(S)$ lie on the unit circle, because they are of form $r_m(\alpha \mathrm{i})$, for some $\alpha \in \mathbb{R}$.

Now, we derive a scalar expression for the error of the approximation

$$\mathrm{e}^S \approx r_m(S).$$

Consider the matrix function

$$E_m(S) = \mathrm{e}^S - r_m(S).$$

The eigenvalues of $E_m(S)$ are given by $E_m(\alpha \mathrm{i}) = \mathrm{e}^{\alpha \mathrm{i}} - r_m(\alpha \mathrm{i})$, for some $\alpha \in \mathbb{R}$. Since $[E_m(S)]^\top = E_m(-S)$, it is easy to prove that the matrix $E_m(S)$ is normal (that is, it commutes with its transpose). Note that the 2-norm of a normal matrix satisfies $\|X\| = \rho(X)$, where $\rho(X)$ denotes the spectral radius of X, i.e., the largest absolute value of the eigenvalues of X ([16, p. 365]). Therefore the absolute Padé error

$$\|E_m(S)\| = \|\mathrm{e}^S - r_m(S)\|,$$

can be written as

$$\|E_m(S)\| = \max_{\alpha \,\mathrm{i} \in \sigma(S)} |\mathrm{e}^{\alpha \mathrm{i}} - r_m(\alpha \,\mathrm{i})|$$

($\sigma(S)$ stands for the spectrum of S). Now consider the real function of real variable

$$f_m(\alpha) = |\mathrm{e}^{\alpha \,\mathrm{i}} - r_m(\alpha \,\mathrm{i})|.$$

This function is even, for all $\alpha \in \mathbb{R}$, because

$$\begin{aligned} f_m(-\alpha) &= |\mathrm{e}^{-\alpha \,\mathrm{i}} - r_m(-\alpha \,\mathrm{i})| \\ &= \left|\overline{\mathrm{e}^{-\alpha \,\mathrm{i}} - r_m(-\alpha \,\mathrm{i})}\right| \\ &= \left|\overline{\mathrm{e}^{-\alpha \,\mathrm{i}}} - \overline{r_m(-\alpha \,\mathrm{i})}\right| \\ &= f_m(\alpha) \end{aligned}$$

but is not monotone. However, there exists $\eta \in \mathbb{R}$ such that f is concave up in the interval $[-\eta, \ \eta]$ and increasing in $[0, \ \eta]$. It has a minimum at $\alpha = 0$ (see Fig. 1 for the particular case f_7). Therefore, if $\|S\| \leq \eta$, the maximal value of f_m on the spectrum of S is attained at $\alpha = \rho(S) = \|S\|$ and consequently

$$\|E_m(S)\| = f_m(\|S\|) = \left|\mathrm{e}^{\|S\|\,\mathrm{i}} - r_m(\|S\|\,\mathrm{i})\right|. \tag{2}$$

This result allows us to understand the behavior of the absolute Padé error and to compute its exact value by means of a scalar function.

We note that the relative Padé error

$$\frac{\|\mathrm{e}^S - r_m(S)\|}{\|\mathrm{e}^S\|}$$

and the absolute Padé error

$$\|\mathrm{e}^S - r_m(S)\|$$

coincide because $\|\mathrm{e}^S\| = 1$ (recall that e^S is orthogonal).

Table 1
Largest values of $\|S\|$ satisfying $f_m(\|S\|) \approx u$.

m	4	7	9	13
$\|S\|$	0.1	1.0	2.0	5.5

Table 2
Numerical experiments for Algorithm 1.

	$\|S\|$	k	Error
Matrix 1	32.03	6	6.36e−015
Matrix 2	88.74	7	8.87e−015
Matrix 3	370.30	9	4.94e−014

One important feature of the error formula (2) is that it provides a criterion to find the minimum number of scalings needed to guarantee the accuracy of Padé approximants. Table 1 displays the approximates for the largest values allowed for $\|S\|$ to guarantee an absolute error near the unit roundoff ($u = 2^{-52} \approx 1.1 \times 10^{-16}$). We only consider the cases $m = 4, 7, 9, 13$ because these Padé approximants have optimal properties with respect to the computational cost (see [6]).

Based on the previous discussion, we suggest the following algorithm for computing the exponential of a skew-symmetric matrix, using the scaling and squaring together with Padé approximants. One uses the Padé approximant of order 7,

$$r_7(x) = p_7(x)/p_7(-x),$$

where

$$p_7(x) = x^7 + 56x^6 + 1512x^5 + 25200x^4 + 277200x^3 + 1995840x^2 + 8648640x + 17297280$$

because it states a good compromise between simplicity and minimization of the computational cost.

Algorithm 1. Given $S \in so(n)$, this algorithm computes e^S.

1. Find the smallest positive integer k such that $\frac{\|S\|}{2^k} \leq 1.0$ (see Table 1);
2. $B = \frac{S}{2^k}$;
3. $B_1 = B^2,\ B_2 = B_1B_1,\ B_3 = B_1B_2$
   $P_1 = 17297280\,I + 1995840\,B_1 + 25200\,B_2 + 56\,B_3$
   $P_2 = B(8648640\,I + 277200\,B_1 + 1512\,B_2 + B_3)$;
4. Solve the linear system with n right-hand sides
   $$(P_1 - P_2)\,r_7(B) = P_1 + P_2$$
   to compute $r_7(B)$;
5. $e^S \approx [r_7(B)]^{2^k}$.

We have optimized the computation of $r_7(B)$ by using the method of [12, p. 574] to compute the polynomials P_1 and P_2 and by avoiding the explicit computation of the inverse $(P_1 - P_2)^{-1}$. Computing $r_7(B)$ requires 4 matrix multiplications plus the solution of a linear system with n right-hand sides. The overall cost of the algorithm is about $2\left(4 + k + \frac{4}{3}\right) n^3$ flops.

Since A is normal, the effects of rounding errors in the squaring phase (step 5) do not represent a difficulty. According to the discussion in [12, p. 576], the rounding error that contaminates the computation of the powers $[r_7(B)]^{2^k}$ is of order

$$\gamma \approx u\|e^S\| = u.$$

In addition, whenever $\|S\| \leq 1$, the condition number of $P_1 - P_2$ in the linear system in step 4 is small.

To illustrate the algorithm above we have generated three randomized skew-symmetric matrices in Matlab, by subtracting to a matrix its transpose. Matrix 1 is 7×7, Matrix 2 is 20×20 and Matrix 3 is 50×50. The results are displayed in Table 2, where k denotes the number of scalings and the `error` concerns to the absolute (or relative) error after the squaring phase (step 5). This error has been estimated by the formula

$$\begin{aligned}\left\| e^S - [r_7(B)]^{2^k} \right\| &= \max_{\lambda \in \sigma(S)} \left| e^\lambda - \left[r_7(\lambda/2^k)\right]^{2^k} \right| \\ &\approx \left| e^{\|S\|i} - \left[r_7(\|S\|i/2^k)\right]^{2^k} \right|.\end{aligned}$$

We note that the matrix under the norm in the left-hand side is normal.

We have also tested the three matrices above with Algorithm 2.1 in [6]. The results are quite similar to those of our algorithm. For matrices 2 and 3 our algorithm requires one less matrix multiplication, which is due to the property that the 2-norm of any matrix can be much smaller than the 1-norm. This means that Algorithm 2.1 in [6] approximately minimizes the number of scalings. Although both algorithms work with different errors (we have used direct error analysis while [6] used inverse), it is interesting to observe that the estimative of the number of scalings uses similar values (compare the values of Table 1 with the values of θ_n in [6, Table 2.1]).

Table 3
Cost of computing Padé approximants to the matrix logarithm.

m	2	3	4	5	6	7	8	9	10	11	12	13
Cost	1M + 2L	2M + 2L	3M + 2L		4M + 2L		5M + 2L		6M + 2L			

4. Computing the logarithm in $SO(n)$

Padé approximants method together with the inverse scaling and squaring is a widely used method for computing the logarithm of a matrix with no eigenvalues on $\mathbb{R}_0^-$. In this section, we restrict this method to orthogonal matrices, having in mind possible simplifications on the standard method. Using some results previously stated in [8], we will show that the number of square roots (the bulk of expense of the method) can be bounded to a maximum of two, thus minimizing the cost and increasing the accuracy.

A method with the same purpose was addressed in [11]. Similarly to the exponential case, we do not find that method appropriate to compute the logarithm of a general orthogonal matrix efficiently, though we recognize its theoretical appeal.

The diagonal Padé approximant of order m of the function $f(x) = \log((1-x)/(1+x))$ is the rational function $R_m(x) = P_m(x)/Q_m(x)$, where the polynomials P_m and Q_m are given by the recursive formulae

$$P_{m+1} = P_m - \frac{m^2x^2P_{m-1}}{4m^2-1}$$
$$Q_{m+1} = Q_m - \frac{m^2x^2Q_{m-1}}{4m^2-1},$$

with the initial conditions $P_0 = 0$, $P_1 = 2x$, $Q_0 = 1$, $Q_1 = 1$. Polynomials in numerator (respectively, in the denominator) have only odd (resp., even) powers.

Let A be a matrix with no nonpositive eigenvalues and let $B = (A-I)(A+I)^{-1}$. If $\|B\|$ is sufficiently close to zero then

$$\log A \approx R_m(B) = P_m(B)\ [Q_m(B)]^{-1}.$$

The approximates given by $R_m(B)$ become more accurate as the order m increases. However, the computational cost and the size of its coefficients also increase. For convenience we restrict m to the set of integers between 2 and 13. Table 3 displays the computational cost of $R_m(B)$, where M stands for the cost of a matrix multiplication and L for the cost of solving a linear system with n right-hand sides. Any Padé approximant involves the solution of two linear system with n right-hand sides: one for the computation of B, which requires the solution of the matrix equation $(A+I)B = A-I$, and the other one for computing $R_m(B)$; this explains the number $2L$ in Table 3. Regarding the computational cost of each Padé approximant, there is no advantage of using, for instance, approximants of orders 6 or 8.

An important feature of Padé approximants to the logarithm is that they are structure preserving in the sense that they satisfy the skew-symmetry property of the logarithm of an orthogonal matrix.

Since Padé approximants are only accurate when $\|B\|$ is sufficiently small, one needs to compute square roots to bring it close to zero. This is the reason why Padé approximants are combined with the inverse scaling and squaring technique [17]. The following steps give a summarized description of this technique; A is a given matrix with no eigenvalues on $\mathbb{R}_0^-$.

1. $B_k = (A^{1/2^k} - I)(A^{1/2^k} + I)^{-1}$; compute k_0 square roots so that $\|B_{k_0}\| \leq \epsilon < 1$.
2. Approximate $\log(A^{1/2^{k_0}}) \approx R_m(B_{k_0})$, where R_m denotes the Padé approximant to $f(x) = \log((1-x)/(1+x))$ of order m.
3. Recover $\log A$ by means of the identity

$$\log A = 2^{k_0} \log\left(A^{1/2^{k_0}}\right).$$

It is the number of square roots in step 1 that guarantees the accuracy of the approximation in step 3. When the Padé error is estimated by upper bounds, poor estimates may lead to the computation of unnecessary square roots. Fortunately, this does not occur for orthogonal matrices, because the Padé error can be given exactly by the scalar function

$$\|\log(Q) - R_m(B)\| = \left|2\arctan(\|B\|) - \hat{R}_m(\|B\|)\right|, \tag{3}$$

where Q is an orthogonal matrix with no negative eigenvalues, $B = (Q-I)(Q+I)^{-1}$, with $\|B\| < 1$, and $\hat{R}_m$ is the Padé approximant of order m of the function $2\arctan(x)$ (see [8, Th. 7.5]). This enable us to find the right values of $\|B\|$ that ensure the required accuracy of Padé approximants. Table 4 lists the largest values of $\|B\|$ that guarantee an absolute Padé error close to the unit roundoff. We can observe that if $\|B\| > 0.5$ one has to compute square roots of Q; otherwise we would have to permit $m > 13$. In the lines below we prove that there is no need to compute more than two square roots.

Table 4
For each m, b_m denotes the largest value of $\|B\|$ for which the error (3) is less than u.

m	3	5	7	9	13
b_m	0.01	0.06	0.16	0.27	0.5

If λ is an eigenvalue of Q then $\lambda = e^{\theta i}$, for some $\theta \in]-\pi, \pi[$. Let $\tilde{\theta}$ be the largest positive angle of the eigenvalues of Q. Since the eigenvalues of $Q - I$ are of the form $\lambda = e^{\theta i} - 1$ and the eigenvalues of $B = (Q - I)(Q + I)^{-1}$ are of the form $\lambda = (e^{\theta i} - 1)/(e^{\theta i} + 1)$, with $\theta \in]-\pi, \pi[$, it is straightforward to prove that

$$\|Q - I\| = 2\sin\left(\frac{\tilde{\theta}}{2}\right) \quad \text{and} \quad \|B\| = \tan\left(\frac{\tilde{\theta}}{2}\right).$$

The latter identity shows that $\|B\|$ can be arbitrarily large. However, if we compute one square root then

$$\|Q^{1/2} - I\| = 2\sin\left(\frac{\tilde{\theta}}{4}\right) \quad \text{and} \quad \|B_1\| = \|(Q^{1/2} - I)(Q^{1/2} - I)^{-1}\| = \tan\left(\frac{\tilde{\theta}}{4}\right);$$

computing one more square root yields

$$\|Q^{1/4} - I\| = 2\sin\left(\frac{\tilde{\theta}}{8}\right) < 0.77 \quad \text{and} \quad \|B_2\| = \|(Q^{1/4} - I)(Q^{1/4} - I)^{-1}\| = \tan\left(\frac{\tilde{\theta}}{8}\right) < 0.42.$$

Since $\|B_2\| < 0.42 < 0.5$, Table 4 shows that it is possible to choose an m such that the Padé absolute error $\|\log Q - R_m(B)\|$ does not exceed the unit roundoff.

The computation of square roots is accessible in Matlab by means of the function `sqrtm`, which implements a method based on the Schur decomposition of a matrix. However, here we use a different approach. As proved in [8], the sequence

$$Y_{k+1} = \frac{Y_k + Y_k^{-T}}{2}, \qquad Y_1 = \frac{Q + I}{2}$$

converges quadratically to $Q^{1/2}$, the principal square root of Q. Two important properties of this iteration are the following: it has nice stability properties and converges to the closest orthogonal matrix to $(Q + I)/2$. This latter property is important for the sake of preserving the structure. Since $Q^{1/2}$ is orthogonal, we must compute approximates that are nearly orthogonal.

Based on the previous discussion we suggest the following algorithm for computing the logarithm of an orthogonal matrix with no negative eigenvalue.

Algorithm 2. Let $Q \in O(n)$ with no negative eigenvalue.

1. By computing square roots, find the smallest $k_0 \in \{0, 1, 2\}$ such that $\|I - Q^{1/2^{k_0}}\| < 0.77$;
2. Find B_{k_0} by solving the linear system with n right-hand sides
$$(Q^{1/2^{k_0}} + I)\, B_{k_0} = Q^{1/2^{k_0}} - I;$$
3. Compute $\|B_{k_0}\|$ and find the smallest $m \in \{3, 5, 7, 9, 13\}$ such that
$$\|B_{k_0}\| \le b_m;$$
(see Table 4)
4. $\log Q \approx 2^{k_0} R_m(B_{k_0})$.

5. Exponentials and logarithms in $SE(n)$

Let $T = \begin{bmatrix} S & \mathbf{u} \\ \mathbf{0} & 0 \end{bmatrix} \in se(n)$. This $(n+1) \times (n+1)$ matrix is (2×2)-block upper triangular whose exponential is given by

$$e^T = \begin{bmatrix} e^S & V\mathbf{u} \\ \mathbf{0} & 1 \end{bmatrix},$$

where $V = \sum_{k=0}^{\infty} \frac{S^k}{(k+1)!} = \int_0^1 e^{St}\, dt$. A natural procedure for computing e^T would be the separated computation of the $(1, 1)$ and $(1, 2)$ blocks. The problem of computing e^S is solved, because S is skew-symmetric. On the other hand, the matrix V could be computed by partial sums or quadratures. Unfortunately, the computation of V would increase considerably the global computational cost. For this reason we analyze the direct application of Padé approximants with scaling and squaring.

According to [13], the accuracy of Padé approximation is guaranteed whenever the norm of each diagonal block of T (S and $\mathbf{0}$) is sufficiently close to zero. Rephrasing, the number of scalings needed to compute e^T does not depend on the norm of the whole matrix T but instead on the norm of its diagonal blocks. This means that Algorithm 1 can be used to compute e^T with one or two trivial modifications. The computational cost is slightly increased just because S is $n \times n$ while T is $(n+1) \times (n+1)$.

For the particular case $n = 3$, which has many applications in robotics, there are several closed formulae (see for instance, [18] and [2, p. 42]) that can be used for computing exponentials in $se(3)$.

Now we focus on the computation of the logarithm of a matrix in $SE(n)$. This problem is easy to solve if $n = 3$. Indeed, let

$$R = \begin{bmatrix} Q & \mathbf{v} \\ \mathbf{0} & 1 \end{bmatrix} \in SE(n),$$

where $R \in SO(3)$, $\mathbf{v} \in \mathbb{R}^{3\times 1}$, and assume that Q has no real negative eigenvalue. The logarithm of R is given by

$$\log R = \begin{bmatrix} \log Q & \mathbf{w} \\ \mathbf{0} & 0 \end{bmatrix} \in se(n),$$

for some $\mathbf{w} \in \mathbb{R}^{3\times 1}$. Denoting the angle associated with R by

$$\theta = \arccos\left(\frac{\mathrm{trace}(R) - 2}{2}\right) \neq 0,$$

we can use the Lagrange–Hermite interpolation formula (see [1]) to conclude that some algebraic calculations give rise to the closed formula

$$\log R = (R - I) + (aI + bR)(R - I)^2,$$

where

$$a = \frac{(1+\cos\theta)(\sin\theta - \theta\cos\theta)}{2\sin^3\theta}$$
$$b = \frac{(1+\cos\theta)(\sin\theta - \sin(2\theta) + \theta\cos(2\theta))}{2\sin^3\theta}.$$

(For a different closed formula for logarithms in $SE(3)$, see [2, p. 414].)

For a general n, we may use Algorithm 2 with minor changes. The matrix B is now given by

$$B = (R - I)(R + I)^{-1} = \begin{bmatrix} (Q - I)(Q + I)^{-1} & \tilde{\mathbf{w}} \\ \mathbf{0} & 0 \end{bmatrix},$$

for some $\tilde{\mathbf{w}} \in \mathbb{R}^{n\times 1}$.

Similarly to the exponential case, it is the norm of the diagonal blocks of B that determines the accuracy of Padé approximants and not the norm of the entire matrix (see [19]). Therefore, the Padé approximant used to approximate $\log Q$ can also be used to compute $\log R$.

6. Conclusions

Estimating the right number of matrix scalings (resp. matrix square roots) in the scaling and squaring (resp. inverse scaling and squaring) method for the matrix exponential (resp. matrix logarithm) has been a very hard question whenever we deal with a general matrix. However, for the particular case of skew-symmetric (resp. orthogonal) matrices, some error analysis enabled us to overcome this problem. In our point of view there is an important gap in the literature in what concerns the effective computation of exponentials in $so(n)$ and logarithms in $SO(n)$. We believe that our analysis and improvements summarized in algorithms 1 and 2 contribute to fill this gap. As shown before, our analysis also has implications on the computation of exponentials in $se(n)$ and logarithms in $SE(n)$, and consequently also in many engineering applications.

Acknowledgement

The second author's work was partially supported by project DENO/FCT-PT (PTDC/EEA-ACR/67020/2006).

References

[1] N.J. Higham, Funtions of Matrices — Theory and Computation, SIAM, Philadelphia, 2008.
[2] R. Murray, Z. Li, S. Sastry, A Mathematical Introduction to Robotic Manipulation, CRC Press, Boca Raton, FL, 1994.
[3] F.C. Park, B. Ravani, Bézier curves on Riemannian manifolds and Lie groups with kinematics applications, ASME J. Mech. Des. 117 (1) (1995) 36–40.

[4] F.C. Park, B. Ravani, Smooth invariant interpolation of rotations, ACM Trans. Graph. 16 (3) (1997) 277–295.
[5] M. Zefran, V. Kumar, C.B. Croke, On the generation of three-dimensional rigid body motions, IEEE Trans. Automat. Control 14 (4) (1998) 576–589.
[6] N.J. Higham, The scaling and squaring method for the matrix exponential revisited, SIAM J. Matrix Anal. Appl. 26 (4) (2005) 1179–1193.
[7] C. Moler, C. Van Loan, Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later, SIAM Rev. 45 (1) (2003) 3–49.
[8] J.R. Cardoso, C.S. Kenney, F.S. Leite, Computing the square root and logarithm of a real *P*-orthogonal matrix, Appl. Numer. Math. 46 (2003) 173–196.
[9] R.W. Brockett, Robotic manipulators and the product of exponentials formulas, in: P.A. Fuhrman (Ed.), Mathematical Theory of Networks and Systems, Springer-Verlag, 1984.
[10] P. Crouch, G. Kun, F. Silva Leite, De Casteljau algorithm for cubic polynomials on the rotation group, in: Proceedings of 2nd Portuguese Conf. on Automatic Control, Coimbra, 9–11 September, 1996, vol. II, pp. 547–552.
[11] J. Gallier, D. Xu, Computing exponentials of skew-symmetric matrices and logarithm of orthogonal matrices, Int. J. Robotics Automation 18 (2003) 10–20.
[12] G. Golub, C. Van Loan, Matrix Computations, 3rd ed., Johns Hopkins University Press, Baltimore and London, 1996.
[13] L. Dieci, A. Papini, Padé approximation for the exponential of a block triangular matrix, Linear Algebra Appl. 25 (2004) 858–899.
[14] R.C. Ward, Numerical computation of the matrix exponential with accuracy estimated, SIAM J. Numer. Anal. 14 (4) (1977) 600–610.
[15] E.B. Saff, R.S. Varga, On the zeros and poles of Padé Approximants to e^z, Numer. Math. 25 (1975) 1–14.
[16] P. Lancaster, M. Tismenetsky, The Theory of Matrices — with Applications, 2nd ed., Academic Press, Orlando, USA, 1985.
[17] C. Kenney, A.J. Laub, Condition estimates for matrix functions, SIAM J. Matrix Anal. Appl. 10 (1989) 191–209.
[18] F.C. Park, J. Bobrow, A recursive algorithm for robot dynamics using Lie groups, Proc. IEEE Int. Conf. Robotics Automation 2 (1994) 1535–1540.
[19] J.R. Cardoso, F.S. Leite, Gregory and Padé error estimates for the logarithm of block triangular matrices, Appl. Numer. Math. 56 (2006) 253–267.